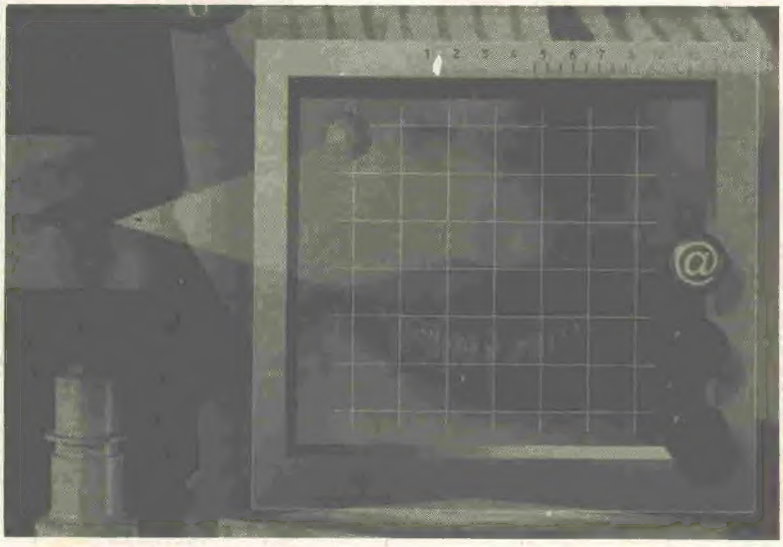
สุทธิศักดิ์ พงศ์ธนาพาณิช

เปิดโลกมัลติมีเดียด้วย VB/Win ตอน ปรับแต่ง effect ของไฟล์ .wav ด้วย WAVETOOL.DLL (ตอนที่ 1)



ไฟล์ไลบรารี WAVETOOL.DLL ช่วยให้การสร้างเทคนิคต่าง ๆ ด้วย VB/Win ทำงานรวดเร็วขึ้น

บทความตอนที่ผ่าน ๆ มา ผู้เขียนได้กล่าวถึงวิธีการเล่นไฟล์ .wav ด้วยฟังก์ชันวินโดว์ API มาแล้ว โดยกล่าวถึงรายละเอียดทั้งวิธีการใช้กลุ่มของฟังก์ชันระดับสูงและฟังก์ชันระดับล่าง (low-level function) ซึ่งผู้อ่านจะเห็นว่าโดยอาศัยกลุ่มฟังก์ชันระดับล่างเราสามารถที่จะตบแต่งเทคนิคการเล่นกลับไฟล์ .wav ได้หลาย ๆ อย่าง เช่น การปรับระดับความดังของการเล่นกลับ (volume level) การเพิ่มหรือลดความเร็ว (pitch change) หรือแม้กระทั่งการสร้างเสียงสะท้อน (echo) เป็นต้น แต่เนื่องจากในการสร้างเทคนิคเหล่านี้ต้องอาศัยการประมวลผลทุกไบต์ของข้อมูลของไฟล์ .wav ดังนั้น ถ้าหากเราเขียนโค้ดส่วนนี้ด้วย VB/Win ก็จะทำให้การทำงานช้าลงไปมากทีเดียว เพราะ VB/Win ไม่สามารถสร้างโค้ดเนทีฟได้โดยตรง ด้วยเหตุนี้ผู้เขียนจึงได้ทำการพัฒนาไฟล์ไลบรารี WAVETOOL.DLL ขึ้นมา ซึ่งไฟล์ไลบรารีดังกล่าวจะสามารถช่วยให้การสร้างเทคนิคข้างต้นด้วย VB/Win เป็นไปได้อย่างรวดเร็วเป็นอย่างมาก (โดยการทดสอบปรากฏว่าเร็วกว่าโค้ดของ VB/Win ประมาณ 2-5 เท่า) และทุก ๆ ฟังก์ชันก็คอมแพตทิเบิลกับ VB/Win 100 เปอร์เซ็นต์ แต่เนื่องจากไฟล์ไลบรารี WAVETOOL.DLL เขียนขึ้นโดยอาศัยกลุ่มฟังก์ชันด้านมัลติมีเดียระดับล่าง ถ้าหากการทำงานไม่ถูกต้องที่ขอให้ผู้อ่านกรุณาแจ้งข้อผิดพลาดกลับมายังผู้เขียนตามที่อยู่ดังนี้ ตู้ ป.ณ.35 ปทฝ. ดินแดง กรุงเทพฯ 10407 ผู้เขียนก็จะได้ทำการปรับปรุงไฟล์ไลบรารีต่อไป และในอนาคตอันใกล้ผู้เขียนก็จะทำการอัพเกรดเป็นเวอร์ชัน 32 บิต สำหรับวินโดวส์ 95 ซึ่งผู้เขียนก็จะแจกฟรีในวารสารไมโครฯต่อไปครับ

ไฟล์ไลบรารี WAVETOOL.DLL ที่มากับบทความตอนนี้เป็นเวอร์ชัน 1.00.01 ก่อนที่จะรันโปรแกรมตัวอย่าง ผู้อ่านต้องคัดลอกไฟล์ดังกล่าวไปไว้ในไดเรคทอรี system ของวินโดวส์เสียก่อน ซึ่งปกติก็จะเป็นไดเรคทอรี c:\windows\system จากนั้นก็สามารถรันโปรแกรมตัวอย่างได้ตามปกติ

ไฟล์ไลบรารี WAVETOOL.DLL

ไฟล์ไลบรารี WAVETOOL.DLL จะประกอบด้วยฟังก์ชันสำหรับการเล่นกลับไฟล์ .wav และการอ่านรายละเอียดด้านความสามารถของอุปกรณ์การเล่นกลับไฟล์ .wav ทั้งด้าน input และ output ซึ่งผู้เขียนก็จะขอแยกฟังก์ชันทั้งหมดออกเป็น 3 กลุ่มหลัก ๆ ตามลักษณะการเรียกใช้งานดังต่อไปนี้

1. ฟังก์ชันทั่วไป ซึ่งจะประกอบด้วยฟังก์ชันที่รายงานเวอร์ชันและการสงวนลิขสิทธิ์ ดังนี้

```
wtGetCopyright
wtGetStrVersion
wtGetVersion
```

2. ฟังก์ชันสำหรับการอ่านรายละเอียดด้านความสามารถของอุปกรณ์การเล่นกลับไฟล์ .wav ทั้งด้าน input และ output และรายงานข้อความแสดงข้อผิดพลาดที่เกิดขึ้นกับไฟล์ไลบรารี ดังนี้

```
wtErrorMsg
wtGetCurrPlayPosition
wtGetInDevInfo
wtGetInDevNum
wtGetOutDevInfo
wtGetOutDevNum
wtGetVersionMMSys
```

3. ฟังก์ชันด้านการเล่นกลับและสร้างเทคนิคสำหรับการเล่นกลับไฟล์ .wav

```
wtGetTimeFormat
wtOpenWaveFile
wtPauseWave
wtPlayReady
wtResumeWave
wtPlayWave
wtSetEcho
wtSetVolume
wtStopWave
wtWaveFormat
wtCloseWaveFile
```

และนอกจากนี้ยังมีค่าคงที่และข้อมูลโครงสร้างต่าง ๆ ที่ใช้ในการกำหนดเงื่อนไขต่าง ๆ ให้กับฟังก์ชันด้านมัลติมีเดียระดับล่าง ดังต่อไปนี้

- ค่าคงที่สำหรับการกำหนดหน่วยวัด ที่ใช้บอกตำแหน่งการเล่นกลับในปัจจุบัน

```
Public Const TIME_MS = &H1        ' time in milliseconds
Public Const TIME_SAMPLES = &H2   ' number of wave samples
Public Const TIME_BYTES = &H4     ' current byte offset
```

- ค่าคงที่สำหรับการกำหนดระดับเสียง (volume)

```
Public Const DEFAULT_VOLUME_CHANGE = 50
```

- ค่าคงที่สำหรับการกำหนดระยะเสียงสะท้อน (echo)

```
Public Const DEFAULT_ECHO = 50
```

- ค่าคงที่สำหรับกำหนดระยะการยืดเสียงสะท้อน (delay time)

```
Public Const DEFAULT_DELAY = 1000
```

- ข้อมูลโครงสร้างสำหรับดีไวซ์ .wav ด้าน input และ output

```
' WAVEOUTCAPS
Public Type wavetoolOutDevInfo
    wMid                As Integer
    wPid                As Integer
    vDriverVersion      As Single
    sPName              As String * 32
    dwFormats           As Long
    wChannels           As Integer
    dwSupport           As Long
    sSupport(0 To 4)    As String * 64
End Type

Public Type wavetoolInDevInfo
    wMid                As Integer
    wPid                As Integer
    vDriverVersion      As Single
    sPName              As String * 32
    dwFormats           As Long
    wChannels           As Integer
    dwSupport           As Long
    sSupport(0 To 4)    As String * 64
End Type
```

wMid และ wPid

ข้อมูลชนิดเลขจำนวนเต็ม บอกถึงรหัสประจำตัวของผู้ผลิตและรหัสประจำตัวผลิตภัณฑ์ตามลำดับ ซึ่งโดยปกติผู้ผลิตการ์ดเสียงต่าง ๆ จะทำการอัพเดตรหัสของตนลงในซอฟต์แวร์ โดยการแจ้งการลงทะเบียนไปยังหน่วยงาน Multimedia Systems Group ของไมโครซอฟท์ สำหรับข้อมูลในตารางที่ 1 เป็นตัวอย่างของรหัสประจำตัวที่ได้รับการลงทะเบียนและอัพเดตลงในซอฟต์แวร์แล้ว

ตารางที่ 1

รหัสประจำตัวผู้ผลิต (Manufacturer ID)

ค่าคงที่	ตัวเลข	รายละเอียด
MM_MICROSOFT	1	ไดรเวอร์ที่ถูกพัฒนาโดยบริษัทไมโครซอฟท์

รหัสประจำตัวผลิตภัณฑ์ (Product ID)

ค่าคงที่	ตัวเลข	รายละเอียด
MM_MIDI_MAPPER	1	ไมโครซอฟต์ MIDI MAPPER
MM_SNDBLST_MIDIOUT	3	พอร์ตส่ง MIDI ของ Sound Blaster
MM_SNDBLST_MIDIIN	4	พอร์ตรับ MIDI ของ Sound Blaster
MM_SNDBLST_SYNTH	5	ซินธีไซเซอร์ภายในของ Sound Blaster
MM_SNDBLST_WAVEOUT	6	พอร์ตส่ง waveform ของ Sound Blaster
MM_SNDBLST_WAVEIN	7	พอร์ตรับ waveform ของ Sound Blaster
MM_ADLIB	9	ซินธีไซเซอร์ที่คอมแพตทิเบิลของ Ad Lib
MM_MPU401_MIDIOUT	10	พอร์ตส่ง MIDI ของ MPU401
MM_MPU401_MIDIIN	11	พอร์ตรับ MIDI ของ MPU401
MM_PC_JOYSTICK	12	การ์ดควบคุมเกมของบริษัท IBM

vDriverVersion

ข้อมูลชนิดเลขจำนวนเต็มกำหนดหมายเลขเวอร์ชันของดีไวซ์ไดรเวอร์สำหรับอุปกรณ์รับสัญญาณเสียง ซึ่งมีการจัดเก็บในรูปแบบเลขฐานสิบหก 2 ไบต์ โดยที่ไบต์ลำดับสูงจะเป็นหมายเลขหลัก (major

number) และไบต์ลำดับล่างจะเป็นหมายเลขรอง (minor number) เช่น ค่า 200H จะหมายถึงหมายเลขเวอร์ชัน 2.00 เป็นต้น

sPName

ข้อมูลชนิดสตริงมีความยาวเท่ากับ 32 ไบต์ (MAXPNAMELEN) สำหรับจัดเก็บรายละเอียดของผู้ผลิตหรือผลิตภัณฑ์ก็ได้

dwFormats

ข้อมูลชนิดเลขจำนวนเต็ม long integer ที่บอกถึงรูปแบบของการจัดเก็บสัญญาณเสียงมาตรฐานที่อุปกรณ์รับสัญญาณเสียงสนับสนุน ซึ่งโดยปกติอุปกรณ์หนึ่ง ๆ จะสนับสนุนมากกว่า 1 รูปแบบ โดยที่ค่าที่จัดเก็บลงในสมาชิกนี้ จึงจัดเก็บในรูปแบบของค่าแฟลก โดยใช้ตัวปฏิบัติการ Or ดังนั้นในการตรวจสอบค่าของ dwFormats จึงต้องใช้ตัวปฏิบัติการ And ในการทดสอบแต่ละแฟลก ซึ่งรูปแบบมาตรฐานที่เป็นไปได้มีดังตารางที่ 2

ตารางที่ 2

ค่าคงที่	ตัวเลข	รายละเอียด
WAVE_INVALIDFORMAT	&H0	รูปแบบไม่เป็นจริง (โดยส่วนใหญ่หมายถึงการเกิดข้อผิดพลาด)
WAVE_FORMAT_1M08	&H1	รูปแบบ 11.025 kHz, Mono, 8-bit
WAVE_FORMAT_1S08	&H2	รูปแบบ 11.025 kHz, Stereo, 8-bit
WAVE_FORMAT_1M16	&H4	รูปแบบ 11.025 kHz, Mono, 16-bit
WAVE_FORMAT_1S16	&H8	รูปแบบ 11.025 kHz, Stereo, 16-bit
WAVE_FORMAT_2M08	&H10	รูปแบบ 22.05 kHz, Mono, 8-bit
WAVE_FORMAT_2S08	&H20	รูปแบบ 22.05 kHz, Stereo, 8-bit
WAVE_FORMAT_2M16	&H40	รูปแบบ 22.05 kHz, Mono, 16-bit
WAVE_FORMAT_2S16	&H80	รูปแบบ 22.05 kHz, Stereo, 16-bit
WAVE_FORMAT_4M08	&H100	รูปแบบ 44.1 kHz, Mono, 8-bit
WAVE_FORMAT_4S08	&H200	รูปแบบ 44.1 kHz, Stereo, 8-bit
WAVE_FORMAT_4M16	&H400	รูปแบบ 44.1 kHz, Mono, 16-bit
WAVE_FORMAT_4S16	&H800	รูปแบบ 44.1 kHz, Stereo, 16-bit

wChannels

ข้อมูลชนิดเลขจำนวนเต็ม ที่บอกถึงจำนวนแชลแนลที่อุปกรณ์รับสัญญาณเสียงสนับสนุน ซึ่งจะมีค่ามากได้ตั้งแต่ 1 ถึง 2 เท่านั้น โดยที่ 1 หมายถึง โมโน (mono) และ 2 หมายถึง สเตอริโอ (stereo)

dwSupports และ Support (0 To 4)

ข้อมูลชนิดเลขจำนวนเต็ม long ซึ่งบอกถึงความสามารถในการควบคุมเพิ่มเติมของอุปกรณ์การเล่นกลับไฟล์ .wav สำหรับไฟล์ไลบรารี WAVETOOL.DLL จะรายงานข้อความแสดงรายละเอียดการสนับสนุนผ่านทางสมาชิก sSupport(0) ถึง sSupport(4) ตามลำดับ ซึ่งมีรายละเอียดดังนี้

```
sSupport(0)      Pitch Control
sSupport(1)      Playback Rate Control
sSupport(2)      Volume Control
sSupport(3)      Separate Left-Right Volume Control
sSupport(4)      Synchronization
```

ข้อมูลโครงสร้างสำหรับการเล่นกลับไฟล์ .wav ด้วยรูปแบบ PCM

```
' PCM wave format.
Public Type wavetoolWaveFormat
    wFormatTa          As Integer   ' format type
    nChannels          As Integer   ' number of
channels (i.e. mono, stereo, etc.
    nSamplesPerSec     As Long      ' sample rate
    nAvgBytesPerSec    As Long      ' for buffer
estimation
    nBlockAlign        As Integer   ' block size of
data
    wBitsPerSample     As Integer   ' bits per sample
End Type
```

wFormatTag

ข้อมูลชนิดเลขจำนวนเต็ม กำหนดชนิดของรูปแบบ ซึ่งในปัจจุบันได้มีการกำหนดเพียงรูปแบบเดียว ดังนี้

ค่าคงที่	ค่าตัวเลข	รายละเอียด
WAVE_FORMAT_PCM	1	ข้อมูลในรูปแบบ PCM

nChannels

ข้อมูลชนิดเลขจำนวนเต็ม กำหนดจำนวนแชลแนลของข้อมูล โดยที่ข้อมูลโมโนจะใช้เพียง 1 แชลแนล ในขณะที่ข้อมูลสเตอริโอจะใช้ 2 แชลแนล

nSamplesPerSec

ข้อมูลชนิดเลขจำนวนเต็ม long กำหนดอัตราการการสุ่มข้อมูลในหน่วยของอัตราการสุ่มข้อมูลใน 1 วินาที

nAvgBytesPerSec

ข้อมูลชนิดเลขจำนวนเต็ม long กำหนดขนาดของข้อมูลเฉลี่ยในหน่วยของไบต์ สำหรับการขนถ่ายในช่วงเวลา 1 วินาที

nBlockAlign

ข้อมูลชนิดเลขจำนวนเต็ม กำหนดขอบเขตการจัดเรียง (alignment) บล็อกในหน่วยของไบต์ โดยที่ข้อมูลในรูปแบบ PCM จะเรียงข้อมูลตามจำนวนไบต์ที่ใช้ในการสุ่มแต่ละครั้ง เช่น ถ้าหากข้อมูลเป็นแบบสเตอริโอ nBlockAlign ก็จะมีค่าเท่ากับ 4 ไบต์ เป็นต้น

wBitsPerSample

ข้อมูลชนิดเลขจำนวนเต็ม กำหนดจำนวนบิตที่ใช้ในการสุ่มข้อมูล 1 ครั้ง เช่น 8 หมายถึง ใช้ 8 บิตในการสุ่มข้อมูลแต่ละครั้ง เป็นต้น

โดยเพื่อให้งานต่อการใช้งาน ผู้เขียนจึงได้ทำการสร้างตัวแปรชนิดครอบคลุม (global variables) ขึ้นมาชุดหนึ่ง ซึ่งในกรณีทั่วไปก็จะเพียงพอที่จะนำไปประยุกต์ใช้งานกับฟังก์ชันทั้งหมดของไฟล์ไลบรารี WAVEOOL.DLL ดังต่อไปนี้

```
Public VolLevel%, EchoRate%, DelayRate%
```

ส่วนรายละเอียดของฟังก์ชันทั้งหมดของไฟล์ไลบรารี WAVETOOL.DLL มีดังต่อไปนี้

wtGetCopyright

Declaration	Public Declare Function wtGetCopyright Lib "WAVETOOL.DLL" () As String
Description	รายงานชื่อผู้เขียนและการสงวนลิขสิทธิ์ซอฟต์แวร์
Parameters	ไม่มี
Example	Debug.Print "Author : "; wtGetCopyright()
Comment	เนื่องจากฟังก์ชันทั้งหมดของไลบรารีได้ถูกเขียนขึ้นมาสำหรับ VB/Win โดยเฉพาะ ดังนั้นในกรณีที่ฟังก์ชันต้องการตัวแปรชนิดสตริงหรือมีการส่งค่ากลับในรูปแบบสตริง ก็ให้ผู้อ่านใช้ตัวแปรสตริงทั่วไปของ VB/Win เป็นตัวแปรให้กับฟังก์ชันหรือรับค่าจากฟังก์ชันก็ได้ เพราะภายในไลบรารีได้ทำการแปลงรูปแบบของสตริงให้คอมแพตทิเบิลกับสตริงของ VB/Win โดยอัตโนมัติอยู่แล้ว (LPSTR)

wtGetStrVersion

Declaration	Public Declare Function wtGetStrVersion Lib "WAVETOOL.DLL" () As String
Description	รายงานสตริงของหมายเลขเวอร์ชันของไฟล์ไลบรารี WAVETOOL.DLL
Parameters	ไม่มี
Example	p$ = "DLL version : " & wtGetStrVersion() & vbCrLf & _"Author :" & wtGetCopyright() & vbCrLf
Comment	ค่าที่ส่งกลับโดยฟังก์ชันนี้จะบอกถึงหมายเลขเวอร์ชันของไฟล์ไลบรารี SCRTOOL.DLL ในรูปแบบของสตริง เช่น 1.00.01 เป็นต้น ซึ่งเหมาะสำหรับการนำหมายเลขเวอร์ชันไปแสดงผล

wtGetVersion

Declaration	Public Declare Function wtGetVersion Lib "WAVETOOL.DLL" () As Long
Description	รายงานหมายเลขเวอร์ชันของไฟล์ไลบรารี WAVETOOL.DLL
Parameters	ไม่มี
Example	lVer& = wtGetVersion() r$ = Format$(Hex$(lVer&), "00000000") ' major.minor.revision verMajor% = Val("&H" & Mid$(r$, 3, 2)) verMinor% = Val("&H" & Mid$(r$, 5, 2)) verRevision% = Val("&H" & Mid$(r$, 7, 2)) p$ = "DLL version : " & verMajor% & "." & verMinor% & "." & verRevision% & vbCrLf p$ = p$ & "Author : " & wtGetCopyright() & vbCrLf
Comment	ค่าที่ส่งกลับโดยฟังก์ชันนี้จะบอกถึงหมายเลขเวอร์ชันของไฟล์ไลบรารี WAVETOOL.DLL หมายเลขเวอร์ชันที่รายงานด้วยฟังก์ชันนี้ เป็นตัวเลขจำนวนเต็ม long จึงเหมาะสำหรับการนำไปประมวลผลและเปรียบเทียบเท่านั้น

wtErrorMsg

Declaration	Public Declare Function wtErrorMsg Lib "WAVETOOL.DLL" _ (ByVal wtErrCode As Integer) As String
Description	รายงานข้อความแสดงรายละเอียดของข้อผิดพลาดที่เกิดขึ้นในการเรียกใช้งานฟังก์ชันต่าง ๆ ของไฟล์ไลบรารี WAVETOOL.DLL
Parameters	wtErrCode ตัวเลขจำนวนเต็ม แสดงหมายเลขประจำข้อผิดพลาดที่เกิดขึ้น

Example	ErrCode% = wtPlayWave()
	If ErrCode% Then
	MsgBox wtErrorMsg(ErrCode%)
	End If

Comment	จากตัวอย่างข้างต้น โค้ดตัวอย่างบรรทัดแรกเป็นการเรียกใช้ฟังก์ชัน wtPlayWave เพื่อทำการเล่นกลับไฟล์ .wav ซึ่งฟังก์ชันที่มีการทำงาน (action function) โดยส่วนใหญ่ของไฟล์ไลบรารี WAVETOOL.DLL จะมีการส่งกลับเป็นรหัสข้อผิดพลาด โดยถ้าหากมีค่ามากกว่า 0 ก็แสดงว่าเกิดข้อผิดพลาดในการเรียกใช้ฟังก์ชัน ซึ่งในกรณีนี้ผู้อ่านสามารถอ่านรายละเอียดของข้อผิดพลาดได้ โดยการส่งรหัสข้อผิดพลาดดังกล่าวให้กับฟังก์ชัน wtErrorMsg ก็จะได้ข้อความนิยามข้อผิดพลาดทันที

wtGetVersionMMSys

Declaration	Public Declare Function wtGetVersionMMSys Lib "WAVETOOL.DLL" () As String
Description	รายงานหมายเลขเวอร์ชันของซอฟต์แวร์ด้านมัลติมีเดียของระบบปฏิบัติการวินโดวส์
Parameters	ไม่มี
Example	Print "MMSYSTEM version: " & wtGetVersionMMSys() & vbCrLf
Comment	โดยปกติหมายเลขเวอร์ชันที่รายงานจะมีค่าดังต่อไปนี้ 1.0 สำหรับระบบ Multimedia Extension 1.0 1.1 สำหรับวินโดวส์ 3.1x

wtGetInDevNum

Declaration	Public Declare Function wtGetInDevNum Lib "WAVETOOL.DLL" () As Integer
Description	รายงานจำนวนดีไวซ์ด้านอินพุตของรูปแบบ waveform
Parameters	ไม่มี
Example	Print wtGetInDevNum()
Comment	สำหรับเครื่องที่มีระบบมัลติมีเดีย จะสนับสนุนดีไวซ์ด้านอินพุตของรูปแบบ waveform อย่างน้อย 1 ตัวเสมอ

wtGetInDevInfo

Declaration	Public Declare Function wtGetInDevInfo Lib "WAVETOOL.DLL" _ (ByVal wtDevID As Integer, wtInDevInfo As wavetoolInDevInfo) As Integer
Description	รายงานความสามารถที่ดีไวซ์ด้านอินพุตของรูปแบบ waveform สนับสนุน
Parameters	wtDevID หมายเลข ID ของดีไวซ์ด้านอินพุตของรูปแบบ waveform โดยมีค่าตั้งแต่ 0 เป็นต้นไป wtInDevInfo ข้อมูลชนิดโครงสร้าง wavetoolInDevInfo สำหรับจัดเก็บรายละเอียดความสามารถของดีไวซ์
Example	Dim wtInDevInfo As wavetoolInDevInfo ' Get output device. NumsID = wtGetInDevNum() Result = wtGetInDevInfo(NumsID - 1, wtInDevInfo)
Comment	Result = 0 หมายถึง ฟังก์ชันสามารถอ่านความสามารถของดีไวซ์ด้านอินพุตของรูปแบบ waveform ได้สำเร็จ <> 0 หมายถึง เกิดข้อผิดพลาด ซึ่งผู้อ่านสามารถอ่านรายละเอียดข้อผิดพลาดได้จากฟังก์ชัน wtErrorMsg

wtGetOutDevNum

Declaration	Public Declare Function wtGetOutDevNum Lib "WAVETOOL.DLL" () As Integer
Description	รายงานจำนวนดีไวซ์ด้านเอาต์พุตของรูปแบบ waveform
Parameters	ไม่มี
Example	Print wtGetOutDevNum()
Comment	สำหรับเครื่องที่มีระบบมัลติมีเดีย จะสนับสนุนดีไวซ์ด้านเอาต์พุตของรูปแบบ waveform อย่างน้อย 1 ตัวเสมอ

wtGetOutDevInfo

Declaration	Public Declare Function wtGetOutDevInfo Lib "WAVETOOL.DLL" _ (ByVal wtDevID As Integer, wtOutDevInfo As wavetoolOutDevInfo) As Integer
Description	รายงานความสามารถที่ดีไวซ์ด้านเอาต์พุตของรูปแบบ waveform สนับสนุน
Parameters	wtDevID หมายเลข ID ของดีไวซ์ด้านเอาต์พุตของรูปแบบ waveform โดยมีค่าตั้งแต่ 0 เป็นต้นไป wtOutDevInfo ข้อมูลชนิดโครงสร้าง wavetoolInDevInfo สำหรับจัดเก็บรายละเอียดความสามารถของดีไวซ์
Example	Dim wtOutDevInfo As wavetoolOutDevInfo ' Get output device. NumsID = wtGetOutDevNum() Result = wtGetOutDevInfo(NumsID - 1, wtOutDevInfo)
Comment	Result = 0 หมายถึง ฟังก์ชันสามารถอ่านความสามารถของดีไวซ์ด้านเอาต์พุตของรูปแบบ waveform ได้สำเร็จ <> 0 หมายถึง เกิดข้อผิดพลาด ซึ่งผู้อ่านสามารถอ่านรายละเอียดข้อผิดพลาดได้จากฟังก์ชัน wtErrorMsg

wtOpenWaveFile

Declaration	Public Declare Function wtOpenWaveFile Lib "WAVETOOL.DLL" _ (ByVal wtszFileName As String, wtWavFmt As wavetoolWaveFormat) As Integer
Description	เปิดไฟล์ .wav ที่กำหนด
Parameters	wtszFileName ชื่อไฟล์ .wav ที่ต้องการเปิด โดยผู้อ่านสามารถบอกชื่อไฟล์โดยการรวมเอาเส้นทาง (full pathname) ก็ได้ wtWavFmt ข้อมูลโครงสร้างชนิด wavetoolWaveFormat สำหรับจัดเก็บรายละเอียด หรือคุณลักษณะ (specification) ของไฟล์ .wav ที่ต้องการเปิด
Example	Dim wtWavFmt As wavetoolWaveFormat On Error Resume Next CMDialog1.filename = mciFilename$ CMDialog1.Filter = "Wave Files|*.wav" CMDialog1.Flags = cdlOFNFileMustExist CMDialog1.ShowOpen If Err = cdlCancel Then Exit Sub mciFilename$ = CMDialog1.filename ErrCode% = wtOpenWaveFile(mciFilename$, wtWavFmt)
Comment	เมื่อผู้อ่านทำการเปิดไฟล์ .wav ได้สำเร็จแล้ว ผู้อ่านสามารถที่จะอ่านรายละเอียดหรือคุณลักษณะเฉพาะของไฟล์ .wav ที่ถูกเปิดได้ โดยการอ่านข้อมูลจากแต่ละสมาชิกของข้อมูลชนิดโครงสร้าง wavetoolWaveFormat

wtGetTimeFormat

Declaration	Public Declare Function wtGetTimeFormat Lib "WAVETOOL.DLL" _ (ByVal wtwType As Integer) As Integer
Description	รายงานรูปแบบของหน่วยวัด (measurement unit) ที่ใช้บอกตำแหน่งการเล่นกลับไฟล์ .wav ในปัจจุบัน
Parameters	wtwType ค่าตัวเลขที่กำหนดรูปแบบของหน่วยวัดตำแหน่งการเล่นกลับ ซึ่งสามารถมีค่าได้ดังต่อไปนี้ TIME_MS หน่วยวัดในรูปแบบของเวลาเป็นวินาที TIME_SAMPLES หน่วยวัดในรูปแบบของจำนวน sample TIME_BYTES หน่วยวัดในรูปแบบของจำนวนไบต์
Example	' play wave for retieve supported time format. ErrCode% = wtPlayWave() If wtGetTimeFormat(TIME_MS) = TIME_MS Then p$ = p$ & " millisecond" & vbCrLf mnuOptionTimeFormat(0).Enabled = True

```
    mnuOptionTimeFormat_Click 0
  Else
    mnuOptionTimeFormat(0).Enabled = False
  End If
  If wtGetTimeFormat(TIME_SAMPLES) = TIME_SAMPLES Then
    TimeFmt% = TIME_SAMPLES
    p$ = p$ & " wave samples" & vbCrLf
    mnuOptionTimeFormat(1).Enabled = True
    mnuOptionTimeFormat_Click 0
    If Not mnuOptionTimeFormat(0).Checked Then
        mnuOptionTimeFormat_Click 1
    End If
  Else
        mnuOptionTimeFormat(1).Enabled = False
  End If
  If wtGetTimeFormat(TIME_BYTES) = TIME_BYTES Then
    p$ = p$ & " bytes offset" & vbCrLf
        mnuOptionTimeFormat(2).Enabled = True
    If Not mnuOptionTimeFormat(1).Checked Then
        mnuOptionTimeFormat_Click 2
    End If
  Else
    mnuOptionTimeFormat(2).Enabled = False
  End If
  ErrCode% = wtStopWave()
```

Comment จากตัวอย่างข้างต้น ก่อนที่ผู้อ่านจะสามารถอ่านหน่วยวัดตำแหน่งการเล่นกลับได้นั้น ผู้อ่านต้องทำการเปิดไฟล์ .wav ที่ต้องการเสียก่อน และที่สำคัญก็ต้องใช้ฟังก์ชัน wtPlayWave เพื่อกำหนดให้มีการเล่นไฟล์ .wav (แต่ในความเป็นจริงการใช้ฟังก์ชันนี้ครั้งแรกหลังจากที่ทำการเปิดไฟล์ด้วยฟังก์ชัน wtOpenWaveFile จะไม่มีการเล่นกลับไฟล์ .wav) จากนั้นจึงสามารถอ่านหน่วยวัดตำแหน่งการเล่นกลับได้

wtGetCurrPlayPosition

Declaration
```
Public Declare Function wtGetCurrPlayPosition Lib "WAVETOOL.DLL" _
(ByVal wtwType As Integer) As Long
```
Description รายงานตำแหน่งการเล่นกลับไฟล์ .wav ในปัจจุบันโดยใช้หน่วยวัด (measurement unit) ตามที่ถูกกำหนด

Parameters wtwType หมายเลขจำนวนเต็ม กำหนดรูปแบบของหน่วยวัดตำแหน่งการเล่นกลับ

Example
```
CurrPos& = wtGetCurrPlayPosition(TimeFmt%)
Print CurrPos&
```
Comment ไม่มี

wtWaveFormat

Declaration
```
Public Declare Function wtWaveFormat Lib "WAVETOOL.DLL" _
(ByVal wtFormat As Long) As String
```
Description รายงานชื่อของรูปแบบ (format) ของไฟล์ .wav

Parameters wtFormat ค่าตัวเลขที่กำหนดรูปแบบของไฟล์ .wav โดยผู้อ่านสามารถดูรายละเอียดรูปแบบที่สนับสนุนได้จากสมาชิกของข้อมูลโครงสร้าง dwFormats ข้างต้น

Example
```
ErrCode% = wtOpenWaveFile(mciFilename$, wtWavFmt)
' Get information of opened file.
Print "Wave Formats : " & wtWaveFormat(wtWavFmt.wFormatTag) & vbCrLf
```
Comment ไม่มี

: ปรับแต่ง effect ของไฟล์ .wav ด้วย WAVETOOL.DLL (ตอนที่ 1)

wtPlayWave

Declaration	Public Declare Function wtPlayWave Lib "WAVETOOL.DLL" () As Integer
Description	กำหนดให้มีการเล่นกลับไฟล์ .wav ที่ถูกเปิดก่อนหน้านี้
Parameters	ไม่มี
Example	ErrCode = wtPlayWave()
Comment	ErrCode = 0 หมายถึง ฟังก์ชันทำงานได้สำเร็จ <> 0 หมายถึง เกิดข้อผิดพลาดภายในฟังก์ชัน ซึ่งผู้อ่านสามารถอ่านรายละเอียดของข้อผิดพลาดได้จากฟังก์ชัน wtErrorMsg

wtPlayReady

Declaration	Public Declare Function wtPlayReady Lib "WAVETOOL.DLL" () As Integer
Description	รายงานสถานะการเล่นกลับของไฟล์ .wav
Parameters	ไม่มี

Example	If wtPlayReady() Then
	Command1.Caption = "&Play"
	Command2.Caption = "Pa&use"
	Text3.Text = ""
	Else
	CurrPos& = wtGetCurrPlayPosition(TimeFmt%)
	If CurrPos& >= 0 Then
	Text3.Text = CurrPos&
	Else
	Text3.Text = ""
	End If
	End If

Comment	ในขณะที่ดีไวซ์ด้านเอาต์พุตกำลังเล่นกลับไฟล์ .wav อยู่นั้น ผู้อ่านสามารถที่จะตรวจสอบสถานะการเล่นกลับว่าเสร็จสิ้นหรือยังได้จากฟังก์ชันนี้ โดยถ้าหากมีค่าไม่เท่ากับ 0 แสดงว่าการเล่นกลับเสร็จสิ้นสมบูรณ์แล้ว แต่ถ้าหากมีค่าเท่ากับ 0 ก็แสดงว่าดีไวซ์ด้านเอาต์พุตกำลังเล่นกลับไฟล์ .wav

wtPauseWave

Declaration	Public Declare Function wtPauseWave Lib "WAVETOOL.DLL" () As Integer
Description	หยุดการเล่นกลับไฟล์ .wav ชั่วคราว
Parameters	ไม่มี
Example	ErrCode% = wtPauseWave()
Comment	ถ้าหากผู้อ่านต้องการเล่นกลับต่อจากตำแหน่งที่หยุดการเล่นกลับไฟล์ .wav ชั่วคราว ผู้อ่านก็สามารถกระทำได้โดยการเรียกใช้ฟังก์ชัน wtResumeWave ดังที่จะได้กล่าวต่อไป

wtResumeWave

Declaration	Public Declare Function wtResumeWave Lib "WAVETOOL.DLL" () As Integer
Description	เล่นกลับต่อจากตำแหน่งที่หยุดการเล่นกลับไฟล์ .wav ชั่วคราวโดยฟังก์ชัน wtPauseWave
Parameters	ไม่มี
Example	ErrCode% = wtResumeWave()
Comment	ไม่มี

wtStopWave

Declaration	Public Declare Function wtStopWave Lib "WAVETOOL.DLL" () As Integer
Description	หยุดการเล่นกลับไฟล์ .wav
Parameters	ไม่มี
Example	ErrCode% = wtStopWave()
Comment	เมื่อผู้อ่านหยุดการเล่นกลับไฟล์ .wav ด้วยฟังก์ชันนี้แล้ว ผู้อ่านสามารถเริ่มต้นเล่นกลับไฟล์ .wav ได้ใหม่อีกครั้งด้วยฟังก์ชัน wtPlayWave เท่านั้น

wtSetVolume

Declaration	Public Declare Function wtSetVolume Lib "WAVETOOL.DLL" _ (ByVal wtVolume As Integer) As Long
Description	ปรับแต่งระดับความดัง (volume) ของการเล่นกลับไฟล์ .wav
Parameters	wtVolume หมายถึง เลขจำนวนเต็มที่กำหนดเปอร์เซ็นต์การเพิ่มหรือลดระดับความดัง โดยถ้าหากมีค่าน้อยกว่า 0 ก็จะเป็นการลด volume แต่ถ้าหากมีค่ามากกว่า 0 ก็จะเป็นการเพิ่ม volume
Example	CalcBytes = wtSetVolume(VolLevel%)
Comment	ฟังก์ชันนี้จะรายงานกลับเป็นจำนวน sample ที่มีการประมวลผล เนื่องจากการปรับเพิ่มหรือลดระดับความดังของการเล่นกลับนั้น ต้องมีการประมวลผลทุก ๆ ไบต์ของข้อมูล ดังนั้นฟังก์ชันนี้อาจจะใช้เวลาในการประมวลผลมาก ทั้งนี้ก็ขึ้นกับขนาดของไฟล์ .wav ด้วยเช่นกัน ผู้อ่านสามารถที่จะตรวจสอบเวลาในการประมวลผลได้โดยใช้กลุ่มของฟังก์ชันของไฟล์ไลบรารี BENMARK.DLL ตามที่เคยกล่าวมาแล้วในวารสารไมโครฯฉบับที่ 134 ประจำเดือนกันยายน 2539 ครับ สำหรับในโปรแกรมตัวอย่าง WTDEMO นี้ ผู้เขียนได้กำหนดให้ใช้ตัวแปร VolLevel% สำหรับจัดเก็บค่าตัวเลขเปอร์เซ็นต์การปรับระดับความดัง (volume)

wtSetEcho

Declaration	Public Declare Function wtSetEcho Lib "WAVETOOL.DLL" _ (ByVal wtDelay As Integer, ByVal wtEchoGain As Integer) As Long
Description	ปรับ echo ของไฟล์ .wav ที่ถูกเปิดในปัจจุบัน
Parameters	wtDelay หมายถึง ค่าตัวเลขที่กำหนดระยะเวลาที่จะใช้ในการยืดจังหวะการทำ echo แต่ละครั้ง ซึ่งมีหน่วยเป็นมิลลิวินาที (ms) wtEchoGain หมายถึง ค่าตัวเลขกำหนดเปอร์เซ็นต์การเกิด echo
Example	CalcBytes = wtSetEcho(DelayRate%, EchoRate%)
Comment	ฟังก์ชันนี้จะรายงานกลับเป็นจำนวน sample ที่มีการประมวลผล เนื่องจากการทำ echo นั้น ต้องมีการประมวลผลทุก ๆ ไบต์ของข้อมูล ดังนั้นฟังก์ชันนี้อาจจะใช้เวลาในการประมวลผลมาก ทั้งนี้ก็ขึ้นกับขนาดของไฟล์ .wav ด้วยเช่นกัน สำหรับในโปรแกรมตัวอย่าง WTDEMO นี้ ผู้เขียนได้กำหนดให้ใช้ตัวแปร EchoRate% สำหรับจัดเก็บค่าตัวเลขเปอร์เซ็นต์การทำ echo และ DelayRate% สำหรับจัดเก็บค่าตัวเลขการยืดจังหวะการทำ echo

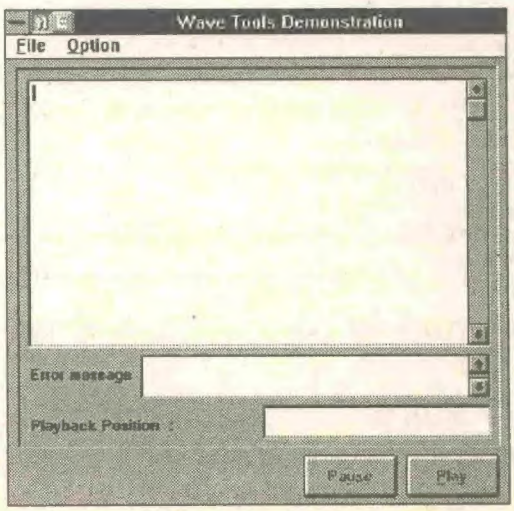
wtCloseWaveFile

Declaration	Public Declare Sub wtCloseWaveFile Lib "WAVETOOL.DLL" ()
Description	ปิดไฟล์ .wav ที่ถูกเปิดก่อนหน้านี้
Parameters	ไม่มี
Example	ErrCode% = wtCloseWaveFile()
Comment	ผู้อ่านต้องทำการปิดไฟล์ .wav ที่ถูกเปิดก่อนหน้านี้ทุกครั้ง ก่อนที่จะทำการเปิดไฟล์ .wav ใหม่อีกครั้ง และห้ามปิดไฟล์ .wav ที่กำลังเล่นกลับหรือปรับแต่งเทคนิคต่าง ๆ ตามข้างต้น

โปรแกรม WTDEMO

เพื่อให้ผู้อ่านได้เข้าใจถึงวิธีการและลำดับการเรียกใช้ฟังก์ชันของไฟล์ไลบรารี WAVETOOL.DLL ผู้เขียนจึงได้สร้างโปรแกรมตัวอย่าง WTDEMO เพื่อประกอบการศึกษาเพิ่มเติม โดยที่โปรแกรม WTDEMO เป็นโปรแกรมอย่างง่ายที่จะแนะนำวิธีการเรียกใช้ฟังก์ชันตามที่กล่าวมาแล้วข้างต้น โดยในบทความฉบับนี้ผู้เขียนก็จะแนะนำเฉพาะวิธีการใช้งานโปรแกรม WTDEMO ส่วนรายละเอียดการทำงานของโปรแกรมและซอร์สโค้ด ผู้เขียนก็ขออนุญาตยกไปไว้ในฉบับหน้าก็แล้วกันนะครับ

เมื่อผู้อ่านทำการคัดลอกไฟล์ไลบรารี WAVETOOL.DLL ไปไว้ในไดเรคทอรี system ของวินโดวส์แล้ว ก็สามารถรันโปรแกรม WTDEMO ได้ทันที โดยเมื่อโปรแกรมถูกโหลดก็จะปรากฏฟอร์มดังในรูปที่ 1 โดยที่โปรแกรมประกอบด้วยเมนูหลัก 2 เมนู ดังนี้ เมนู File และเมนู Options โดยที่แต่ละเมนูจะประกอบด้วยคำสั่งต่าง ๆ ดังรายละเอียดต่อไปนี้

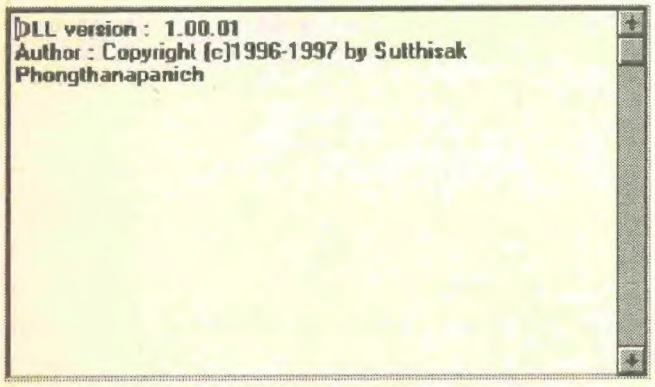


รูปที่ 1 ฟอร์มของโปรแกรม WTDEMO เมื่อถูกโหลดครั้งแรก

เมนู File

คำสั่ง Get DLL information

ทำหน้าที่รายงานรายละเอียดของไฟล์ไบรารี WAVETOOL.DLL เช่น หมายเลขเวอร์ชัน และชื่อผู้เขียนซอฟต์แวร์ โดยเมื่อเลือกคำสั่งนี้ก็จะมีการแสดงรายละเอียดใน TextBox ดังในรูปที่ 2



รูปที่ 2 รายละเอียดของไฟล์ไลบรารี WAVETOOL

คำสั่ง Get Device Information

ทำหน้าที่รายงานคุณสมบัติ (specification) ทั้งหมดของอุปกรณ์ทั้งด้านอินพุตและเอาต์พุตด้านการเล่นกลับไฟล์ .wav นอกจากนี้ยังรายงานหมายเลขเวอร์ชันของซอฟต์แวร์มัลติมีเดียสำหรับระบบปฏิบัติการวินโดวส์อีกด้วย รายละเอียดที่ได้จากคำสั่งนี้จะมีประโยชน์เป็นอย่างมาก เพราะสามารถที่จะตรวจสอบความสามารถที่สนับสนุนโดยอุปกรณ์ด้านอินพุตและเอาต์พุตด้านการเล่นกลับไฟล์ .wav ที่ติดตั้งในเครื่องของผู้อ่าน โดยในรูปที่ 3 แสดงตัวอย่างของรายการที่อุปกรณ์ด้านอินพุตและเอาต์พุตด้านการเล่นกลับไฟล์ .wav สนับสนุน

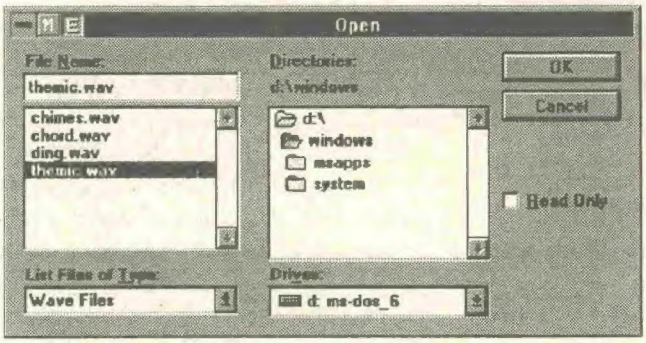
รูปที่ 3 ตัวอย่างความสามารถที่อุปกรณ์การเล่นกลับไฟล์ .wav สนับสนุน

คำสั่ง Initialize Wave Devices

ถ้าหากผู้อ่านพบข้อผิดพลาดในขณะที่กำลังเปิดไฟล์ .wav หรือไม่สามารถเล่นกลับไฟล์ .wav ได้ตามปกติ ก็สามารถที่จะใช้คำสั่งนี้ในการบังคับให้มีการปิดไฟล์ที่ถูกเปิดค้างเอาไว้ก่อนหน้านี้ได้ทันที เนื่องจากไฟล์ .wav นั้นมีอยู่ด้วยกันหลาย ๆ รูปแบบ ดังนั้นถ้าหากไฟล์ที่ถูกเลือกมีรูปแบบที่ไม่สนับสนุนโดยไดรเวอร์ที่ติดตั้งในระบบของผู้อ่านก็จะทำให้เกิดข้อผิดพลาด ซึ่งจะทำให้ไม่สามารถเปิดไฟล์ .wav ที่ต้องการได้ ซึ่งในกรณีนี้อาจจะทำให้ไฟล์ไลบรารี WAVETOOOL.DLL ทำงานไม่ถูกต้อง ดังนั้นผู้เขียนจึงขอแนะนำให้จบการทำงานของโปรแกรมแล้วโหลดใหม่อีกครั้ง สำหรับในกรณีนี้เท่านั้น

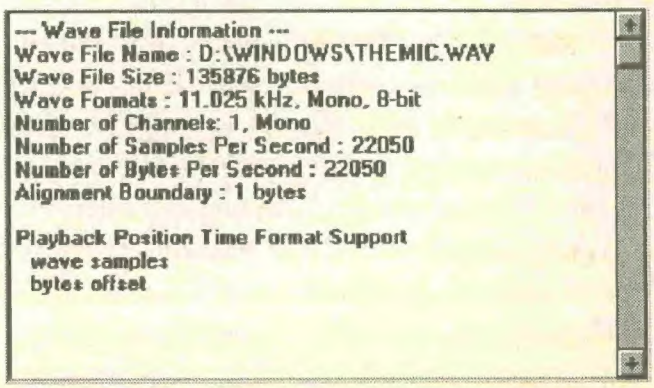
คำสั่ง OpenWave File

ก่อนที่ผู้อ่านจะสามารถเล่นกลับหรือสร้างเทคนิคต่าง ๆ ให้กับไฟล์ .wav ผู้อ่านก็จะต้องทำการเปิดไฟล์ที่ต้องการด้วยคำสั่งนี้เสียก่อน โดยเมื่อเลือกคำสั่งนี้ก็จะปรากฏไดอะล็อกบ็อกซ์ดังในรูปที่ 4



รูปที่ 4 แสดงไดอะล็อกบ็อกซ์สำหรับการเลือกไฟล์ .wav ที่ต้องการ

เมื่อผู้อ่านเลือกไฟล์ .wav ที่ต้องการแล้ว โปรแกรมก็จะทำการเปิดไฟล์และอ่านคุณลักษณะ (characteristics) ของไฟล์ที่ถูกเปิดพร้อมทั้งรายงานเป็นรายละเอียดดังตัวอย่างในรูปที่ 5 โดยจะรายงานขนาดของไฟล์, channel, playback mode, จำนวนยอดคลื่นสุ่มต่อวินาที (sample per second), จำนวนไบต์ต่อวินาที (byte per second), รูปแบบของไฟล์ .wav (wave format) หรือหน่วยวัดตำแหน่งการเล่นกลับในปัจจุบัน



รูปที่ 5 แสดงตัวอย่างคุณลักษณะของไฟล์ themic.wav

ที่สนับสนุน เป็นต้น ซึ่งผู้อ่านสามารถใช้ข้อมูลเหล่านี้ในการประมาณขนาดของไฟล์ .wav ที่จะเกิดขึ้น ถ้าหากมีการอัดเสียงเป็นความยาวในเวลากี่วินาที เช่น ถ้าหากผู้อ่านทำการอัดเสียงลงในไฟล์ .wav ในรูปแบบ 11.025 kHz, mono channel และจัดเก็บด้วยข้อมูลขนาด 8 บิต ซึ่งมีความยาวเท่ากับ 10 วินาที ก็จะกินพื้นที่จัดเก็บทั้งหมดประมาณ 10 x 22050 x 1 = 220,500 ไบต์ เป็นต้น

คำสั่ง Exit

ทำหน้าที่จบการทำงานของโปรแกรม

เมนู Options

คำสั่ง Set Time Format

เป็นคำสั่งที่ใช้กำหนดรูปแบบของหน่วยวัดตำแหน่งการเล่นกลับในปัจจุบัน เพื่อที่จะให้บางฟังก์ชันของไฟล์ไลบรารี WAVETOOL.DLL ทำการรายงานตำแหน่งการเล่นกลับในปัจจุบัน โดยใช้ค่าตามรูปแบบที่กำหนด ซึ่งสามารถกำหนดได้ 3 หน่วยดังนี้ Millisecond, Sample และ Byte

คำสั่ง Set Volume

ทำหน้าที่แสดงบล็อกตามรูปที่ 6 เพื่อให้ผู้อ่านสามารถกำหนดตัวเลขเปอร์เซ็นต์ของระดับเสียงที่ต้องการให้เพิ่มหรือลด โดยถ้าหากค่าของ Volume Rate มีค่ามากกว่า 0 ก็แสดงว่าเป็นการเพิ่มระดับเสียง แต่ถ้าหากมีค่าน้อยกว่า 0 ก็แสดงว่าเป็นการลดระดับเสียง โดยผู้อ่านสามารถที่จะคลิ๊กที่ปุ่มคำสั่ง Default ก็ได้ถ้าหากต้องการค่าปกติที่ตั้งเอาไว้

คำสั่ง Change Volume

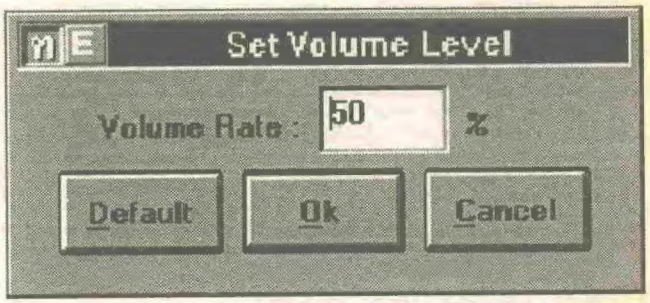
คำสั่งนี้ทำหน้าที่เปลี่ยนแปลงระดับความดังในการเล่นกลับไฟล์ .wav ตามค่าที่กำหนดให้กับตัวแปร VolLevel%

คำสั่ง Set Echo

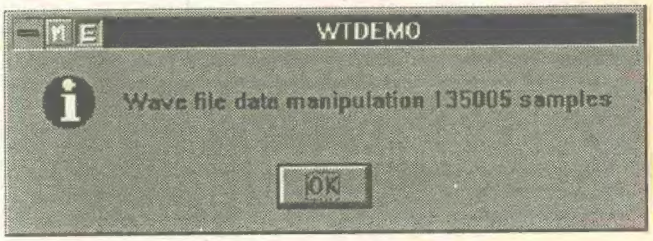
เป็นคำสั่งที่ใช้กำหนดเงื่อนไขการทำ echo โดยจะประกอบด้วยค่า Delay Rate หมายถึงช่วงระยะเวลาที่หน่วงเพื่อให้เกิด echo ซึ่งมีหน่วยเป็นมิลลิวินาที และ Echo Gain Level ซึ่งมีค่าเป็นเปอร์เซ็นต์สำหรับกำหนดอัตราการขยายสัญญาณ echo

คำสั่ง Change Echo

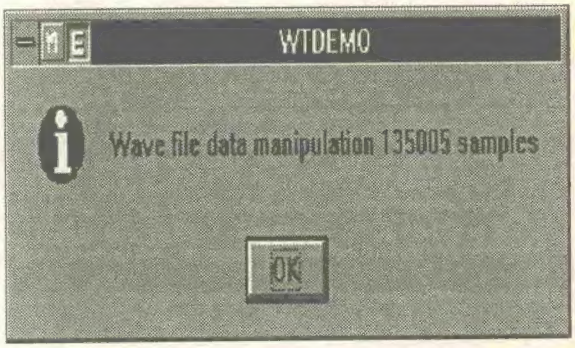
ทำหน้าที่สร้าง echo ให้กับไฟล์ .wav ที่ถูกเปิดก่อนหน้านี้ โดยเมื่อเลือกคำสั่งนี้ไฟล์ไลบรารี WAVETOOL.DLL จะทำการประมวลผลทุก ๆ ไบต์ และเมื่อเสร็จสิ้นขบวนการทำ echo แล้วก็จะปรากฏข้อความเพื่อบอกถึงจำนวน sample ที่ถูกประมวลผลซึ่งเหมือนกับในรูปที่ 7



รูปที่ 6 แสดงบล็อกสำหรับการกำหนดเปอร์เซ็นต์ของระดับความดัง



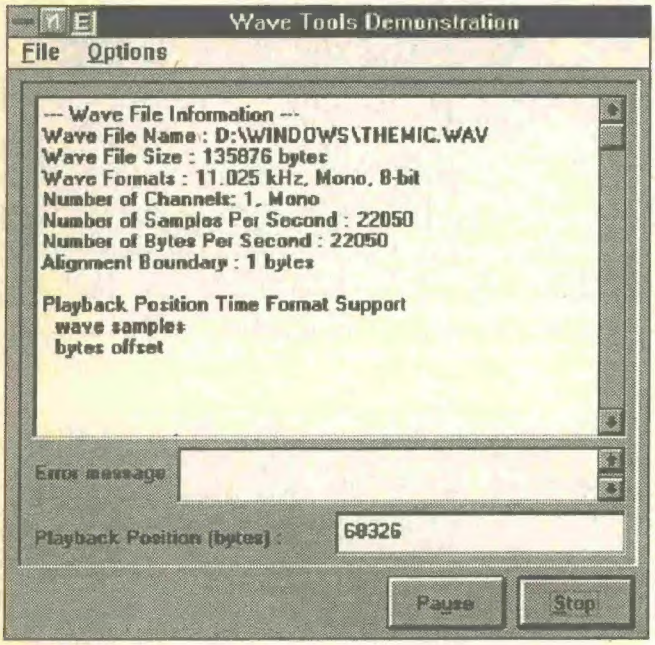
รูปที่ 7 แสดงข้อความบอกถึงจำนวน sample ที่ถูกประมวลผล



รูปที่ 8 แสดงบล็อกสำหรับการกำหนดคุณลักษณะที่ใช้ทำ echo

ซึ่งผู้อ่านจะเห็นว่าโปรแกรม WTDEMO ใช้งานได้ง่ายทีเดียว โดยเมื่อมีการเลือกคำสั่งที่ทำให้เกิดการติดต่อหรือปรับแต่งเทคนิคกับไฟล์ .wav นั้น โปรแกรมก็จะรายงานสถานะการทำงานหรือข้อผิดพลาดที่เกิดขึ้นให้ผู้อ่านได้รับทราบ โดยการแสดงผลลงในช่อง Error Message และในขณะที่มีการเล่นกลับไฟล์ .wav นั้นโปรแกรมก็จะรายงานตำแหน่งการเล่นกลับปัจจุบันลงในช่อง Playback Position ซึ่งจะมีหน่วยตามที่ได้ถูกกำหนดเอาไว้โดยคำสั่ง Set Time Format

และนอกจากนี้ในระหว่างที่มีการเล่นไฟล์นั้น ผู้อ่านสามารถที่จะหยุดการเล่นกลับได้ชั่วคราวโดยการคลิ๊กที่ปุ่มคำสั่ง Pause และคลิ๊กที่ปุ่มคำสั่ง Resume ถ้าหากต้องการให้มีการเล่นกลับต่อเนื่องจากตำแหน่งที่หยุด และคลิ๊กที่ปุ่มคำสั่ง Play เพื่อทำการเล่นกลับไฟล์ .wav จากเริ่มต้น หรือคลิ๊กที่ปุ่มคำสั่ง Stop เพื่อหยุดการเล่นกลับทันที ซึ่งในกรณีที่หยุด



รูปที่ 9 แสดงตำแหน่งการเล่นกลับในช่อง Playback Position

การเล่นกลับด้วยปุ่มคำสั่ง Stop นั้น ผู้อ่านจะไม่สามารถเล่นกลับต่อจากตำแหน่งที่หยุดก่อนหน้านี้ได้เลย นอกจากการเริ่มต้นเล่นกลับจากจุดเริ่มต้นใหม่เท่านั้น โดยการคลิ๊กที่ปุ่มคำสั่ง Play นั่นเอง

สำหรับในบทความตอนต่อไป ผู้เขียนก็จะได้อธิบายถึงซอร์สโค้ดและขั้นตอนการทำงานของโปรแกรม WTDEMO ต่อไปครับ สุดท้ายก็ขอให้ผู้อ่านติดตามบทความในตอนต่อ ๆ ไปครับ พบกันใหม่ในตอนต่อไปสวัสดีครับ

เทคนิคประจำคอลัมน์

ถาม ทำไมจึงเกิด GPF กับ VB/Win?
ตอบ GPF หมายถึง General Protection Fault ซึ่งจะเกิดขึ้นกับ VB/Win ด้วยสาเหตุหลัก ๆ ดังนี้
1. ส่งตัวแปรผิดประเภทให้กับฟังก์ชันวินโดว์ API
2. ใช้ตัวเลือก Byval ในประโยค Declare ผิดตำแหน่ง
3. ในกรณีที่มีการส่งตัวแปรชนิดสตริงให้กับฟังก์ชันวินโดว์ API เพื่อเป็นบัฟเฟอร์สำหรับรับค่ากลับ ถ้าหากขนาดของสตริงเล็กเกินไป ก็สามารถเกิด GPF ได้เช่นกัน
ดังนั้น ถ้าหากเกิด GPF ผู้อ่านก็ควรจะทำการตรวจสอบตามรายการข้างต้นโดยเฉพาะการประกาศและเรียกใช้ฟังก์ชันวินโดว์ API

เวบไซท์ยอดนิยม

เวบไซท์ 100hot.com ได้จัดเรียงอันดับความนิยมจากความถี่ของจำนวนผู้ใช้อินเทอร์เน็ตที่แวะเข้าไปเยี่ยมชมในแต่ละไซท์ คราวนี้ลองตรวจดูว่าคุณแวะเข้าไปครบทุกไซท์หรือยัง ?

1. http://www.yahoo.com
2. http://www.geocities.com
3. http://www.pathfinder.com
4. http://www.warnerbros.com
5. http://www.dccomics.com
6. http://www.extratv.com
7. http://www.babylon5.com
8. http://www.cnn.com
9. http://www.cris.com
10. http://www.usatoday.com
11. http://www.teleport.com
12. http://www.primenet.com
13. http://www.w3.org
14. http://www.unitedmedia.com
15. http://www.webcom.com
16. http://www.cnet.com
17. http://www.search.com
18. http://www.news.com
19. http://www.tiac.net
20. http://www.aol.com
21. http://www.webcrawler.com
22. http://www.xmission.com
23. http://www.well.com
24. http://www.nando.net
25. http://www.lycos.com
26. http://www.pointcom.com
27. http://www.wam.umd.edu
28. http://www.100hot.com
29. http://www.mindspring.com
30. http://www.cbs.com
31. http://www.mcp.com
32. http://www.interlog.com
33. http://www.hooked.net
34. http://www.discovery.com
35. http://www.mca.com
36. http://www.sjmercury.com
37. http://www.whitehouse.gov
38. http://www.apple.com
39. http://www.cs.uit.no
40. http://www.galaxy.einet.net

(อ่านต่อได้ที่ หน้า 236)